ESE535:
Electronic Design Automation

Day 9: February 11, 2013
Placement
(Intro, Constructive)

Today
• 2D Placement Problem
• Partitioning ➔ Placement
• Quadrisection
• Refinement

Placement
• Problem: Pick locations for all building blocks
  – minimizing energy, delay, area
  – really:
    • minimize wire length
    • minimize channel density

Bad Placement
• How bad can it be?
  – Area
  – Delay
  – Energy

Preclass Channel Widths
• Channel Width for Problem 1?

Preclass Channel Widths
• Channel Width for Problem 2?
Bad: Area

- All wires cross bisection
- $O(N^2)$ area
- Good: $O(N)$

Delay

- How bad can delay be?

Bad: Delay

- All critical path wires cross chip
- Delay = $O(|PATH|^2 L_{side})$
  - [$L_{side}$ is $O(N)$]
- Good: $O(|PATH| L_{g})$
- Compare 10ps gates to many nanoseconds to cross chip

Bad: Energy

- All wires cross chip:
  - $O(L_{side})$ long → $O(L_{side})$ capacitance per wire
  - Recall Area → $O(N^2)$
  - So $L_{side}$ → $O(N)$
  - $O(N)$ wires → $O(N^2)$ capacitance
- Good:
  - $O(1)$ long wires → $O(N)$ capacitance

Clock Cycle Radius

- Radius of logic can reach in one cycle (45 nm)
  - 1 Cycle Radius = 10
  - Few hundred PEs
  - Chip side 600-700 PE
  - 400-500 thousand PEs
  - 100s of cycles to cross
Manhattan Distance

- Horizontal and Vertical Routing:
  Manhattan distance
  \[ |X_i - X_j| + |Y_i - Y_j| \]
- Contrast:
  Euclidean distance
  \[ \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \]

Distance

- Can we place everything close?

Illustration

- Consider a complete tree
  - nand2's, no fanout
  - N nodes
- Logical circuit depth?
- Circuit Area?
- Side Length?
- Average wire length between nand gates? (lower bound)

“Closeness”

- Try placing “everything” close

<table>
<thead>
<tr>
<th>Manhattan Distance</th>
<th>Places</th>
<th>Transitive Fanin</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>64</td>
</tr>
<tr>
<td>4</td>
<td>4d</td>
<td>4d</td>
</tr>
</tbody>
</table>

Alternate Wire Length Illustration

- Consider a cut width
  \[ F(N) > \sqrt{N} \]
- If optimally place all
  \[ F(N) \text{ producers right next to bisection} \]
  - How many cells deep
    is the producer farthest from the bisection?
Generalizing Interconnect Lengths

\[ BW = F(N) > \sqrt{N} \]

- Large cut widths imply long wires

Placement Problem Characteristics

- Familiar
  - NP Complete
  - local, greedy not work
  - greedy gets stuck in local minima

Constructive Placement

Basic Idea

- Partition (bisect) to define halves of chip
  - minimize wire crossing
- Recurse to refine
- When get down to single component, done

Adequate?

- Does recursive bisection capture the primary constraints of two-dimensional placement?

Problems

- Greedy, top-down cuts
  - maybe better pay cost early?
- Two-dimensional problem
  - (often) no real cost difference between H and V cuts
- Interaction between subtrees
  - not modeled by recursive bisect
Interaction

Example

Ideal split (not typical)

"Equivalent" split ignoring external constraints
Practically -- makes all H cuts also be V cuts

Problem

• Need to keep track of where things are
  – outside of current partition
  – include costs induced by above
• ...but don’t necessarily know where things are
  – still solving problem

Improvement: Ordered

• Order operations
• Keep track of existing solution
• Use to constrain or pass costs to next subproblem

Improvement: Ordered

• Order operations
• Keep track of existing solution
• Use to constrain or pass costs to next subproblem
• Flow cut
  – use existing in src/sink
  – A nets = src, B nets = sink
Improvement: Ordered

- Order operations
- Keep track of existing solution
- Use to constrain or pass costs to next subproblem
- Flow cut
  - use existing in src/sink
  - A nets = src, B nets = sink
- FM: start with fixed, unmovable nets for side-biased inputs

Improvement: Constrain

- Partition once
- Constrain movement within existing partitions
- Account for both H and V crossings
- Partition next
  - (simultaneously work parallel problems)
  - easy modification to FM

Constrain Partition

Solve AB and CD concurrently.

Improvement: Quadrisect

- Solve more of problem at once
- Quadrisect:
  - partition into 4 bins simultaneously
  - keep track of costs all around

Quadrisect

- Modify FM to work on multiple buckets
- k-way has:
  - k(k-1) buckets
  - |from|×|to|
  - quad → 12
- reformulate gains
- update still O(1)
Recursion

- Keep outside constraints
  - (cost effects)
- Problem?
  - Don’t know detail place
- What can we do?
  - Model as at center of unrefined region

Option: Terminal Propagation

- Abstract inputs as terminals
- Partition based upon
- Represent cost effects on placement/refinement decisions

Option: Refine

- Keep refined placement
- Use in cost estimates

Problem

- Still have ordering problem
- What is the problem?
- Earlier subproblems solved with weak constraints from later
  - (cruder placement estimates)
- Solved previous case by flattening
- Why might not be satisfied with that?
- In extreme give up divide and conquer
- Alternative?

Iterate

- After solve later problems
- “Relax” solution
- Solve earlier problems again with refined placements (cost estimates)
- Repeat until converge

Iteration/Cycling

- General technique to deal with phase-ordering problem
  - what order do we perform transformations, make decisions?
  - How get accurate information to everyone
- Still basically greedy
Refinement

- Relax using overlapping windows
- Deal with edging effects
- Huang & Kahng claim 10-15% improve
  - cycle
  - overlap

Possible Refinement

- Allow unbalanced cuts
  - most things still work
  - just distort refinement groups
  - allowing unbalance using FM quad-section looks a bit tricky
  - gives another 5-10% improvement

Runtime

- Each gain update still $O(1)$
  - (bigger constants)
  - so, FM partition pass still $O(N)$
- $O(1)$ iterations expected
- assume $O(1)$ overlaps exploited
- $O(\log(N))$ levels
- Total: $O(N \log(N))$
  - very fast compared to typical annealing
  - (annealing next time)

Quality: Area

Gordian-L: Analytic global placer
DOMINO: network flow placer

<table>
<thead>
<tr>
<th>Case</th>
<th>GOHELU</th>
<th>DOMINO</th>
<th>QUAD</th>
<th>GOHELU</th>
<th>DOMINO</th>
</tr>
</thead>
<tbody>
<tr>
<td>prim1</td>
<td>10598</td>
<td>10590</td>
<td>10298</td>
<td>2.8%</td>
<td>1.5%</td>
</tr>
<tr>
<td>prim2</td>
<td>45094</td>
<td>43875</td>
<td>44479</td>
<td>3.3%</td>
<td>1.6%</td>
</tr>
<tr>
<td>inst7</td>
<td>409300</td>
<td>412764</td>
<td>380154</td>
<td>12.5%</td>
<td>8.9%</td>
</tr>
<tr>
<td>inst3</td>
<td>1120100</td>
<td>1049872</td>
<td>970498</td>
<td>12.5%</td>
<td>7.5%</td>
</tr>
<tr>
<td>fract</td>
<td>400</td>
<td>383</td>
<td>380</td>
<td>5.0%</td>
<td>0.8%</td>
</tr>
<tr>
<td>C1098</td>
<td>1858</td>
<td>1767</td>
<td>1820</td>
<td>1.5%</td>
<td>3.6%</td>
</tr>
<tr>
<td>C5315</td>
<td>6220</td>
<td>5992</td>
<td>6185</td>
<td>6.6%</td>
<td>4.4%</td>
</tr>
<tr>
<td>C7284</td>
<td>8744</td>
<td>8220</td>
<td>8012</td>
<td>5.5%</td>
<td>3.2%</td>
</tr>
<tr>
<td>s1438</td>
<td>2334</td>
<td>2388</td>
<td>2355</td>
<td>2.0%</td>
<td>2.6%</td>
</tr>
<tr>
<td>s1438</td>
<td>2908</td>
<td>2958</td>
<td>2906</td>
<td>7.8%</td>
<td>3.4%</td>
</tr>
<tr>
<td>s3578</td>
<td>8609</td>
<td>8182</td>
<td>8208</td>
<td>4.7%</td>
<td>0.4%</td>
</tr>
<tr>
<td>s2234</td>
<td>14848</td>
<td>14803</td>
<td>14848</td>
<td>6.7%</td>
<td>3.3%</td>
</tr>
<tr>
<td>s12927</td>
<td>31284</td>
<td>29645</td>
<td>29163</td>
<td>9.0%</td>
<td>6.1%</td>
</tr>
<tr>
<td>s15860</td>
<td>31020</td>
<td>32541</td>
<td>31295</td>
<td>9.2%</td>
<td>5.1%</td>
</tr>
<tr>
<td>struct</td>
<td>4160</td>
<td>3967</td>
<td>4156</td>
<td>0.0%</td>
<td>3.8%</td>
</tr>
<tr>
<td>biomed</td>
<td>34677</td>
<td>33712</td>
<td>33787</td>
<td>2.6%</td>
<td>0.2%</td>
</tr>
<tr>
<td>area</td>
<td>10648</td>
<td>92355</td>
<td>93867</td>
<td>0.2%</td>
<td>3.6%</td>
</tr>
<tr>
<td>avg1</td>
<td>100650</td>
<td>97825</td>
<td>101520</td>
<td>-1.3%</td>
<td>-2.5%</td>
</tr>
<tr>
<td>avg2</td>
<td>100650</td>
<td>97825</td>
<td>101520</td>
<td>-1.3%</td>
<td>-2.5%</td>
</tr>
</tbody>
</table>

Quality: Delay

- Weight edges based on criticality
  - Periodic, interleaved timing analysis

Uses

- Good by self
- Starting point for simulated annealing
  - speed convergence
- With synthesis (both high level and logic)
  - get a quick estimate of physical effects
  - (play role in estimation/refinement at larger level)
- Early/fast placement
  - before willing to spend time looking for best
- For fast placement where time matters
  - FPGAs, online placement?
Summary

• Partition to minimize cut size
• Additional constraints to do well
  – Improving constant factors
• Quadrisection
• Keep track of estimated placement
• Relax/iterate/Refine

Big Ideas:

• Potential dominance of interconnect
• Divide-and-conquer
• Successive Refinement
• Phase ordering: estimate/relax/iterate

Admin

• Assignment 3 out
• No Office Hours Tuesday
• No class Wednesday
• Reading for next Monday
  – Online (JSTOR): classic paper on Simulated Annealing
• Drop Day is Friday
  – ...I will try to make some grading progress while I travel...